

"BookShelf"

**A PROJECT REPORT SUBMITTED TO
THE NATIONAL INSTITUTE OF ENGINEERING, MYSURU**

(An Autonomous College)



ESTD : 1946

In partial fulfillment of the requirements for Project work (Database Laboratory), fifth semester

Bachelor of Engineering

in

Computer Science and Engineering

Submitted by

Sudarshan R. Kulkarni (4NI16CS104)

Sumanth S.A (4NI16CS105)

Suraj Vashista B.K (4NI16CS108)

Under the Guidance of

Mr.G.Ramesh
Assistance Professor
Dept. of CS&E
NIE, Mysuru

Mr.Suhas S
Assistance Professor
Dept. of CS&E
NIE, Mysuru

**DEPARTMENT OF COMPUTER SCIENCE AND
ENGINEERING**

THE NATIONAL INSTITUTE OF ENGINEERING

Mysore-570 008

2018-2019

**DEPARTMENT OF COMPUTER SCIENCE AND
ENGINEERING**
THE NATIONAL INSTITUTE OF ENGINEERING



ESTD : 1946

CERTIFICATE

This is to certify that the project work entitled “BookShelf” is a work carried out by Sudarshan R. Kulkarni (4NI16CS104), Sumanth S.A (4NI16CS105) and Suraj Vashista B.K (4NI16CS108) in partial fulfillment for the project work (Database Laboratory), fifth semester, Computer Science & Engineering, The National Institute of Engineering (Autonomous Institution under Visvesvaraya Technological University, Belgaum) during the academic year 2018-2019. It is certified that all corrections and suggestions indicated for the Internal Assessment have been incorporated in the report deposited in the department library. The project work report has been approved in partial fulfillment as per academic regulations of The National Institute of Engineering, Mysuru.

Internal Guides

Professor & Head

Mr.G.Ramesh
Assistance Professor
Dept. of CS&E
NIE, Mysuru

Mr.Suhas S
Assistance Professor
Dept. of CS&E
NIE, Mysuru

Dr. H D Phaneendra
Professor and Head
Dept. of CS&E
NIE, Mysuru

ACKNOWLEDGEMENT

We would like to take this opportunity to express out profound gratitude to all those people who were directly or indirectly involved in the completion of this project. We thank each and every one who encouraged us in every possible way.

We would like to thank our principal Dr.G.Ravi for letting us to be the part of this prestigious institution and letting us to explore abilities to the fullest.

We would also like to extend our sincere gratitude to Dr.H.D.Phaneendra, our H.O.D for being the source of inspiration and instilling an enthusiastic spirit in us throughout the process of project making.

We would like to express our heartfelt gratitude towards our project guides Mr.G.Ramesh and Mr.Suhas S for their constant guidance ,valuable knowledge and experience.

We are thankful to our family and friends for their constant support, inspiration and encouragement, for their helping hand and all those who supported us directly or indirectly in our academic process.

Thank you

-Sudarshan R. Kulkarni

-Sumanth S.A

-Suraj Vashista B.K

Table of Contents

<u>Contents</u>	<u>Page</u>
Abstract	
1. Introduction	1
2. System Analysis	3
2.1 Existing Systems	4
2.2 Proposed System Explanation	4
2.3 System Requirements	5
3. System Design	6
3.1 Tables Used in Database	6
3.2 ER Diagram	10
4. System Implementation	11
4.1 Design	11
4.2 Backend	13
4.3 Frontend	14
4.4 How It Works	14
5. System Testing	15
5.1 Test Cases	15
6. Snapshots	16
Conclusion and Future Enhancements	
References	

List of Figures

<u>Figure No.</u>	<u>Description</u>	<u>Page</u>
3.1	ER Diagram	10
6.1	Login page	16
6.2	Signup page	17
6.3	User Homepage	17
6.4	Lending Section	18
6.5	Add New Book	18
6.6	Reading Section	19
6.7	Request A Book	19
6.8	User Profile	20

Abstract

BookShelf is a project aimed at providing enthusiastic readers the means to discover and read new books as well as share their own books with others. It is a platform for these readers to read to their heart's content free of cost.

The users may create a profile for themselves for lending or borrowing a book in the BookShelf, wherein, the lenders' can upload the details of their books which will be stored in the BookShelf for as long as they wish to lend it. And the readers may look-up a book of their interest on the platform for borrowing. Based on the availability of the book in the reader's vicinity, the BookShelf provides a means for the reader to connect to one of the lenders' who possesses the book of interest.

Furthermore, the BookShelf provides the readers with recommendations by keeping a track of ratings of books provided by different readers. This will allow the reader to discover popular books based on the rating given by different users.

Lastly, the platform does not force a single role (lender/receiver) on any user as he/she may be a lender and a receiver at the same time.

Also, the admin interface is kept exclusive to the developers to add information of new books as well as keep track of all the transactions that occur on the platform.

Chapter - 1

Introduction

Buying a book every time one wants to read isn't feasible, nor realistic for all people. Neither can one go door to door to see if the people in his vicinity have the book that he so desires. This project aims at remedying these issues by providing a platform where the person can not only search for people who have that book, but he can also lend off the books that he may currently possess.

This allows for a system where a user of the system is both a lender and a reader at the same time. Thus, saving him the hassle of buying books very often, and also helps in putting his old books to some good use by lending them off to those in need of it.

New users can directly register to our website by creating an account. The accounts are password protected and will have the option to change the password.

The reader has an option to request the book in need from a particular lender from the given list, which is generated by taking into account the readers location and the search query provided. Once the request gets accepted by the lender, both the end users are provided information regarding each other so as to facilitate communication between them regarding the various nuances of sending and receiving the book.

The lender may upload books that he currently possesses and wishes to lend. He may add any book that is currently present in our vast database of books. If a book that he chose to lend has been requested, the lender is informed of the same through a table notifying him of pending requests. He may choose to either accept or reject this request. Also, there is an option to pause his lends as and when there may be long periods of inactivity by him.

The administrator manages the overall working of the website. He may add new releases to the database. He may also access all transaction information since the beginning of time.

The web interface is built using HTML, CSS and JS. Databases are maintained by SQLITE3 and the website is hosted using the Flask-python framework. Any fairly modern computer with an operating system that supports a web-browser can be used to use the website.

Chapter - 2

System Analysis

System analysis is a detailed study of the various operations performed by a system and their relationships within and outside of the system. Here the key question is - What all problems exist in the present system? What must be done to solve the problem? Analysis begins when a user or manager begins a study of the program using existing system.

During analysis, data collected on various files, decision points and transactions handled by the present system. The commonly used tools in the system are Data Flow-Diagrams, interviews, etc. training experience and common sense are required for collection of relevant information needed to develop the system. The success of the system depends largely on how clearly the problem is defined, thoroughly investigated and properly carried out through the choice of the solution.

A good analysis model should provide not only the mechanism of the problem understanding but also the framework of the solution. The proposed system should be analysed thoroughly in accordance with the needs. System analysis can be categorized into four parts.

1. System planning and initial investigation
2. Information gathering
3. Applying analysis tools for structured analysis
4. Feasibility study

2.1 Existing Systems

There are some web applications which provide similar functionality as our platform, such as

- There is a platform by the name [Hireabook](#) that lets its users rent/sell their books in their neighborhood.
- Another platform called [Hoo.gy](#) allows users to borrow textbooks from friends or strangers for a nominal fee.

2.2 Proposed System Explanation

Our project uses Python 3.6 for the backend. The Flask library provides a hosting service for http and ftp responses. Sqlite3 is used for the database. It is an RDBMS which can be easily integrated with Flask python. These two systems handle the server side. The frontend of our website uses HTML. The Materialize package of CSS and JS components is used to stylize the website. Custom CSS and JS components are also used where necessary.

The Flask Project can be run on a server with network access. This is important because the website grabs font, CSS and icon files from Google's server. The website requires a fairly modern web browser with JS enabled. Details about the user's system requirements is discussed later in the report.

Program or Package	Version Used
Python	3.6.5
Sqlite3	2.5
Flask	1.0.2
HTML	5

Table 2.1

2.3 System Requirements

Server side:

Hardware:

- 1: A dual core CPU @ 2.5GHz
- 2: minimum of 256 MB ram
- 3: 20 GB of storage
- 4: Fast and high bandwidth internet connection
- 5: A router with port forwarding setup for the server

Software:

- 1: Compatible version of any Linux or Windows OS
- 2: Python 3.6
- 3: Django

User side:

Hardware:

- 1: A dual core CPU @ 2.5GHz
- 2: minimum of 1 GB ram
- 3: 20 GB of storage
- 4: Internet connection
- 5: 766p monitor @ 60Hz
- 6: Keyboard and mouse setup

Software:

- 1: Compatible version of any Linux or Windows OS
- 2: A web browser with JS enabled.

Chapter - 3

System Design

3.1 Tables Used in the Database:

user_info:

Fields	Data Types	Relationships
UID	Integer	Primary Key, Autoincrement
name	varchar(50)	Not null
street	varchar(30)	Not null
city	varchar(20)	Not null
state	varchar(20)	Not null
country	varchar(20)	Not null
contact_no	int(10)	Not null
email	varchar(50)	Not null
password	varchar(20)	Not null

Table 3.1

The user_info table (Table 3.1) is used to maintain the information of users that are using our website. It contains UID as its primary key and is auto-incremented to ensure that a new ID is given to every user. This is used to link information in other tables to a particular user. The email and password fields are used to validate the login credentials and uniquely identify a user.

books:

Fields	Data Types	Relationships
ISBN	int(5)	Primary Key
book_name	varchar(40)	Not null
author	varchar(50)	Not null
count	Integer	Default 0
rating	Decimal(1,5)	Default 0

Table 3.2

The books table (Table 3.2) is used to store all the relevant information regarding any book in the database. Each book is uniquely identified by its ISBN. The count field denotes the number of people who have rated that particular book and the rating field gives the average rating out of five ‘starts’ of that book.

lending_section:

Fields	Data Types	Relationships
ID	Integer	Primary Key, Autoincrement
LID	Integer	Foreign Key(user_info)
ISBN	int(5)	Foreign Key(books)
av	int(1)	Not null
RID	Integer	Foreign Key(user_info)
lent_date	Date	Not null
due_date	Date	Not null
transaction_id	Integer	Not null

Table 3.3

The lending_section table (Table 3.3) is used to hold the information of every uploaded book by the users. Each entry is uniquely identified by its ID field which is auto-incremented. It also consists of the LID (lender’s ID) and the RID (reader’s ID) to get their respective information from the user_info table. The av field is used to determine whether the book is in possession of the lender (1), reader (0), has been requested (2) or paused from lending (3).

reading_section:

Fields	Data Types	Relationships
ID	Integer	Primary Key, Autoincrement
RID	Integer	Foreign Key(user_info)
ISBN	int(5)	Foreign Key(books)
due_date	Date	Not null
extn_count	Integer	Default 0
LID	Integer	Foreign Key(user_info)
read_status	Integer	Not null
transaction_id	Integer	Not null

Table 3.4

The reading_section table (Table 3.4) is used to hold the information of current or previous reads of the users. Each entry is uniquely identified by its ID field which is auto-incremented. It also consists of the LID (lender's ID) and the RID (reader's ID) to get their respective information from the user_info table.

The extn_count (extension count) field is used to limit the number of renewals that a reader can make on a book. The transaction_id is used for getting the list of all the transactions by the admin.

incomplete_transaction:

Fields	Data Types	Relationships
transaction_id	Integer	Primary Key, Autoincrement
RID	Integer	Foreign Key(user_info)
LID	Integer	Foreign Key(user_info)
L_ID	Integer	Foreign Key(lending_section)
t_date	Date	Not null

Table 3.5

This table (Table 3.5) is used to temporarily store the information regarding a requested book by a reader indicated by the RID, for which an action has not been taken by the specified lender given by the LID.

The t_date field is used to keep track of date of transaction.

3.2 ER Diagram:

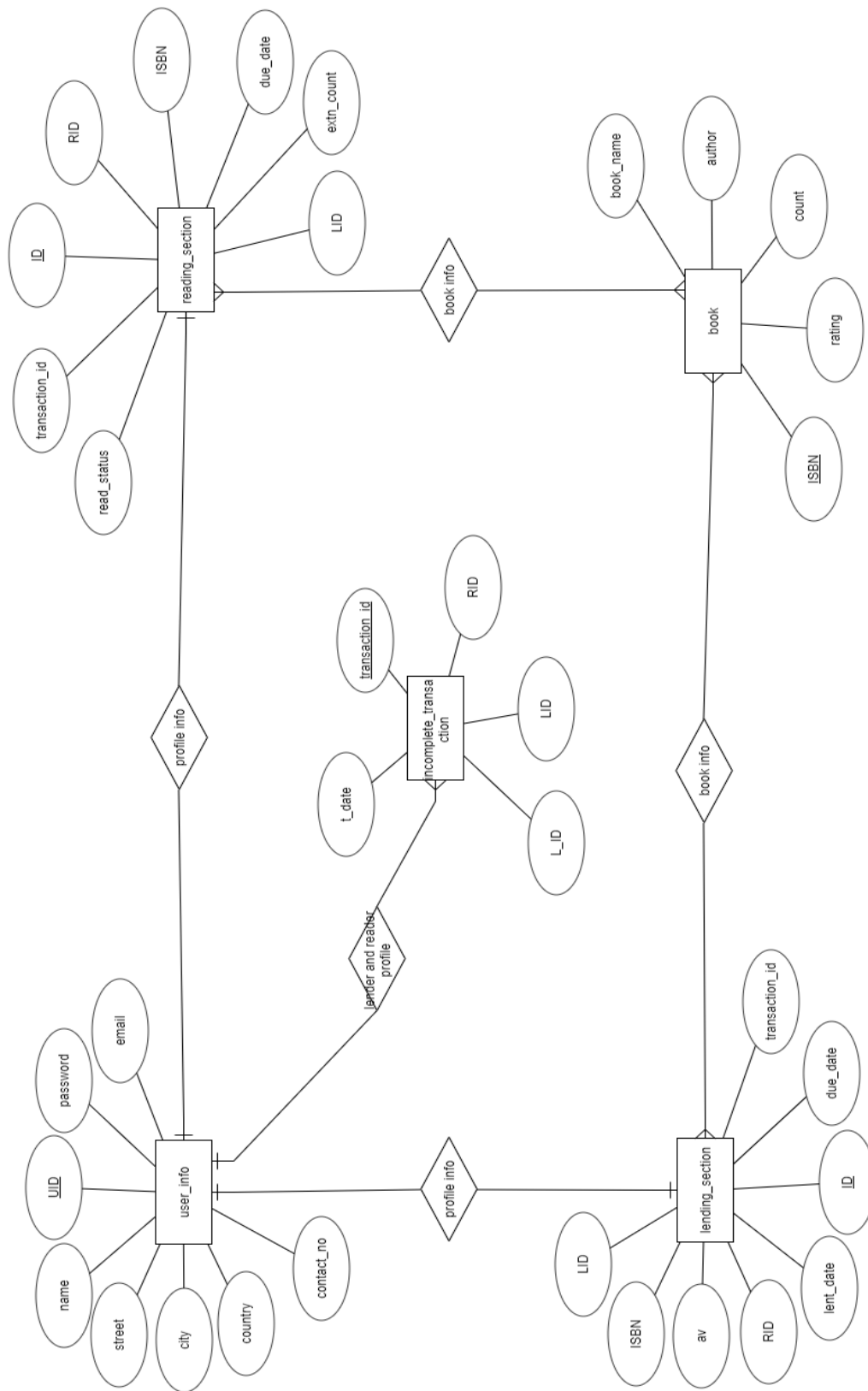


Figure 3.1

Chapter - 4

System Implementation

4.1 Design

As discussed above the website is made using Flask library on Python 3.6. Sqlite3 is used as the database manager. The frontend is made using HTML, CSS and JS. The details of these libraries and languages in given below:

Flask:

Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. Extensions are used for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools.

The microframework Flask is based on the *Pocoo* projects *Werkzeug* and *Jinja2*.

Werkzeug

Werkzeug is a utility library for the Python programming language, a toolkit for Web Server Gateway Interface (WSGI) applications and is licensed under a BSD License. Werkzeug can realize software objects for request, response, and utility functions. It can be used to build a custom software framework on top of it.

Jinja

Jinja is a template engine for the Python programming language and is licensed under a BSD License. Similar to the Django web framework, it provides that templates are evaluated in a sandbox.

SQLite3:

SQLite is a relational database management system contained in C programming library. In contrast to many other database management systems,

SQLite is not a client-server database engine. Rather, it's embedded into the end program.

SQLite is ACID-compliant and implements most of the SQL standard, using a dynamic and weakly typed SQL syntax that does not guarantee the domain integrity.

SQLite is a popular choice as embedded database software for local/client storage in application software such as web browsers. It is arguably the most widely deployed database engine, as it is arguably the most widely deployed database engine, as it is used today by several widespread browsers, operating system, and embedded systems (such as mobile phone), among others. SQLite has bindings to many programming languages.

HTML:

HTML or HyperText Markup Language is the standard mark-up language used to create webpages. The purpose of the web browser is to read HTML documents and compose them into visible or audible web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page. HTML describes the structure of a website semantically along with cues for presentation, making it a mark-up language rather than a programming language.

Web browsers can also refer to cascading style sheets (CSS) to define the looks and layout of text and other material. The W3C maintainer of both the HTML and the CSS standards, encourages the use of CSS over explicit presentational HTML.

CSS:

Cascading style sheets(CSS) is a style sheet language used for describing the presentation of a document written in a mark-up language. Although most often used to set the visual style of web pages and user interface written in HTML and XHTML, the language can be applied to any XML document, including plain XML, SVG and XUL, and is applicable to reading in speech or on other media. Along

with HTML and JavaScript, CSS is a cornerstone technology used by most websites to create visually engaging web pages, user interface for web applications, and user interface for many mobile applications.

CSS is designed primarily to the separation of document content from document presentation, including aspects such as the layout, colours and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple HTML pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

JavaScript:

JavaScript, often abbreviated as JS, is a high-level, dynamic, weakly typed, prototype-based, multi-paradigm, and interpreted programming language. Alongside HTML and CSS, JavaScript is one of the three core technologies of World Wide Web Content production. It is used to make webpages interactive and provide online programs, including video games. The majority of websites employ it, and all modern web programs support it without the need for plug-ins by means of a built-in JavaScript engine.

4.2 Backend:

The backend of the website is written in python using the Flask library. A html page is sent to the browser when a proper URL is entered in the browser. All POST requests are handled with a CSRF token and all temporary data is deleted when it is no longer needed.

4.3 Frontend:

The materialize package is used to stylize the frontend. Elements like popups, scroll effects, autofill, toasts, input verification, button hit effects, etc. are added to improve the look of the websites.

4.4 How it works:

There are three main sections in the website:

- User Login:
 - As a lender a user can use the functions in his lending section to add a new book in his possession to lend, remove a book from the lending service, take an action on any requested book (accept/reject) or pause/resume his lends.
 - As a reader a user can use the functionality in the reading section of the website to search for and request new books (he is provided with the top 5 popular books as rated by other users). He is provided an option for renewing a book for a period of 7 days (which may be done 3 times). He is also given a list of his previous reads.
- Admin Login:

This section is used by the admin to exert control over the database. An admin has the ability to update the database with the information of new books. Also, he may view the list transactions that have taken place since the beginning for further information.

Chapter - 5

System Testing

5.1 Test Cases:

The website was tested with a set or random generated data. The data included a multitude of cases to ensure that all things run as expected. Test cases of leaving the website without logging out or trying to extract user data without a proper login id and/or password is also tested.

The password change mechanism is also tested to ensure that the new password is compliant with the standard of a minimum of 8 characters in length.

The registration forms were also tested with improper entries like entering a string of characters instead of the phone number. If there is any discrepancy in the entered data, the entered form is not processed and a proper error message is sent back to the user.

The lending section was tested for adding new books by the users. The system was tested so as to check its integrity when displaying the valid books or returning a relevant error message if no such book was found to the user.

Also, the remove a book section was checked to allow only those books which have not been lent to anybody to be removed.

The pause functionality was tested to check to allow only those books which were currently in possession of the lender to be paused temporarily from lending.

The reading section was tested for checking the integrity of the software while displaying the information of the requested in the user's vicinity or return a relevant error message if no such book is found.

Chapter - 6

Snapshots

Login

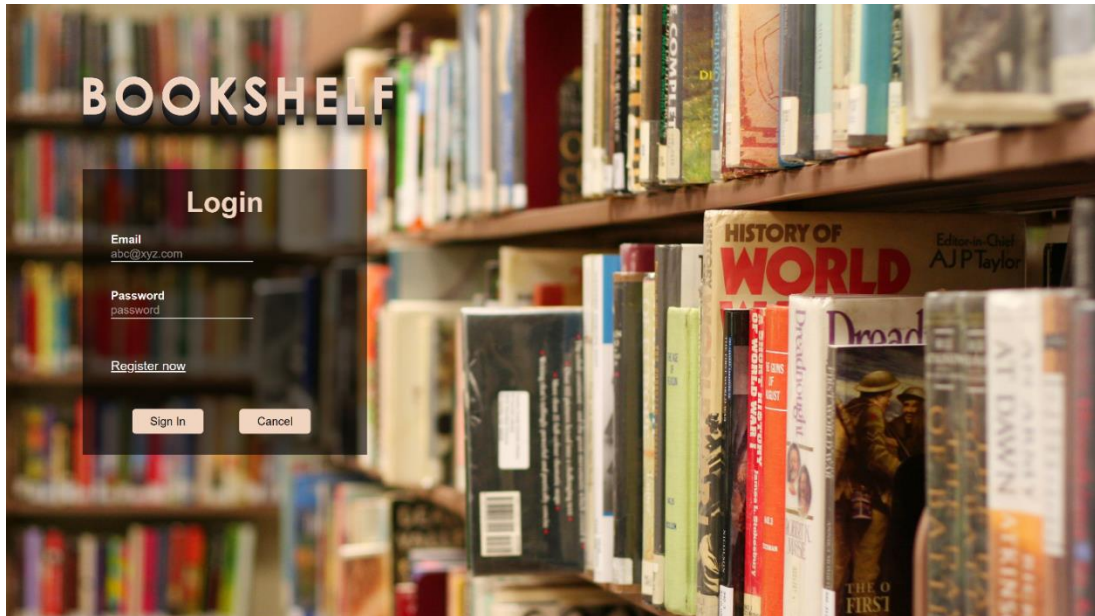
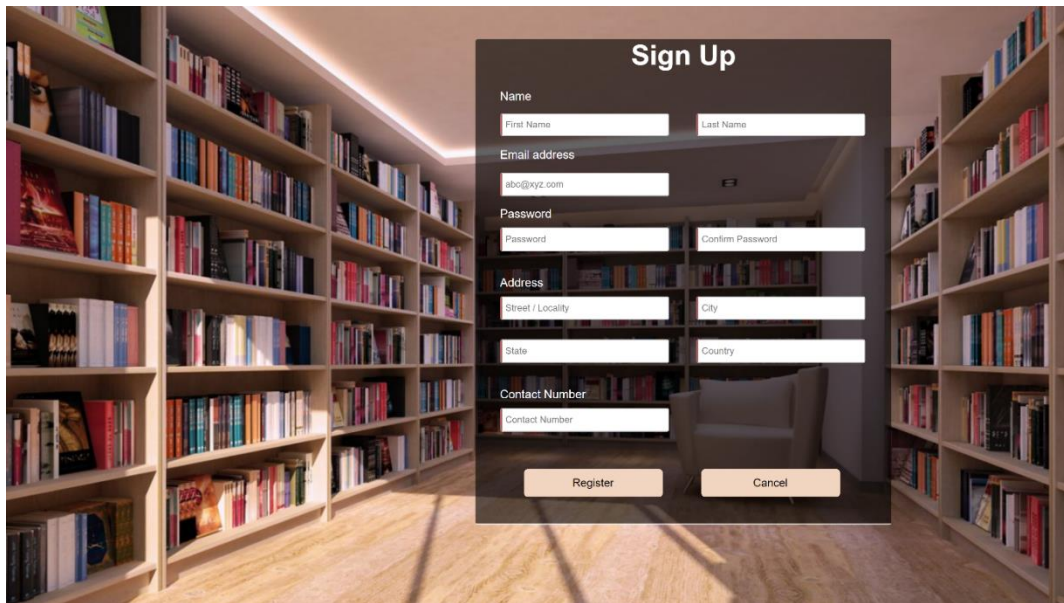


Figure 6.1

Figure 6.1 shows the login page of the website, wherein a user can login to his/her account or create a new account of his own using the “Register now” link.

Signup



Sign Up

Name
First Name Last Name

Email address

Password
Password Confirm Password

Address
Street / Locality City
State Country

Contact Number

Figure 6.2

Figure 6.2 depicts the signup page where a user may enter his details and create an account of his own.

User Home

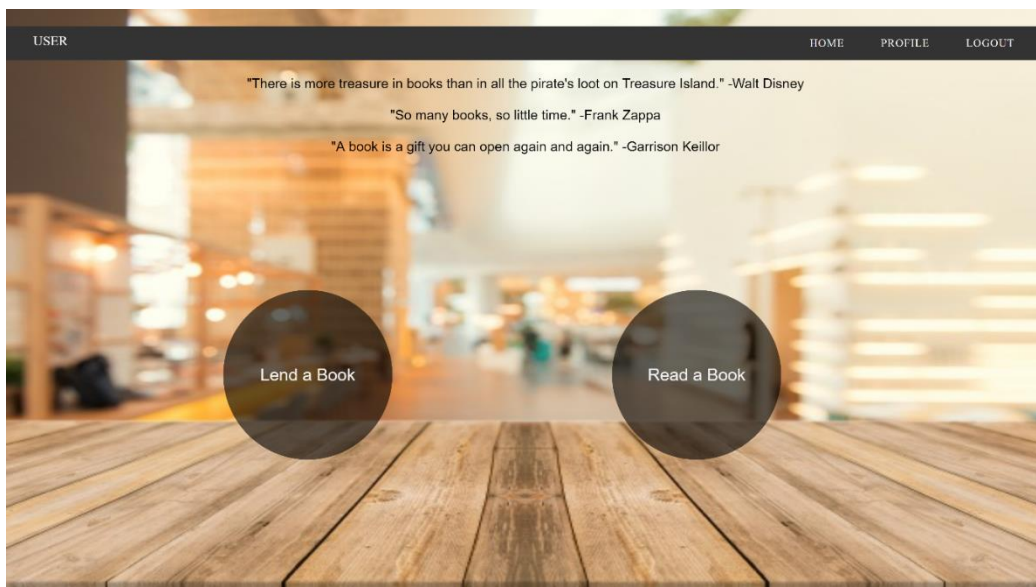


Figure 6.3

Figure 6.3 shows the user's homepage, displayed once he has logged in, which can be used to navigate to his lending or reading sections or profile page.

Lending Section

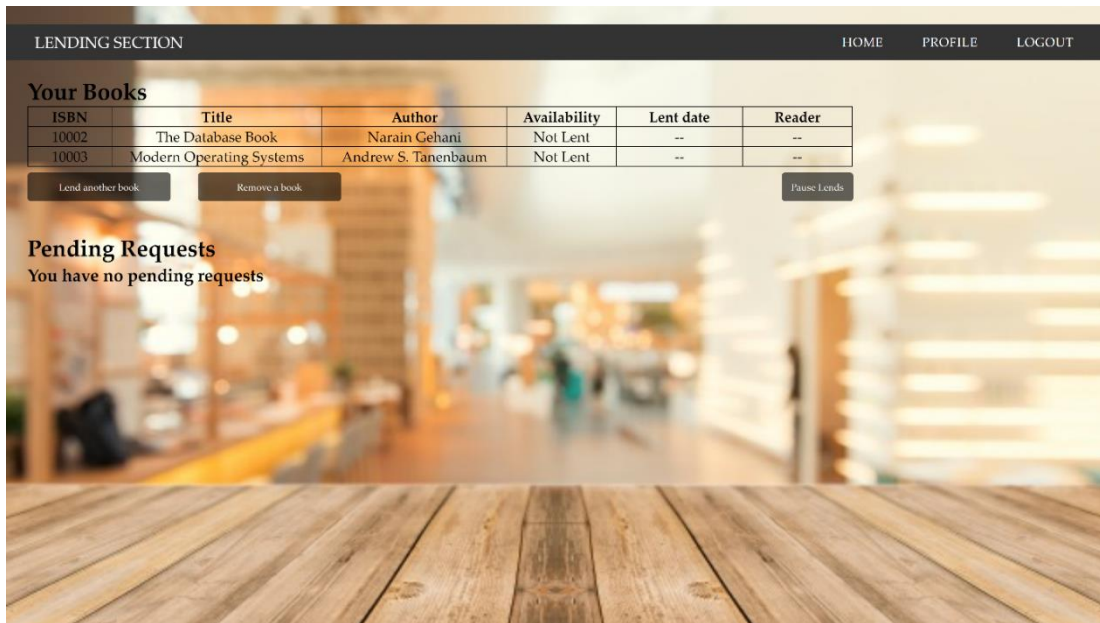


Figure 6.4

Figure 6.4 shows the lending section wherein the user can view the books he has lent or have been requested and take the necessary action.

Add New Book

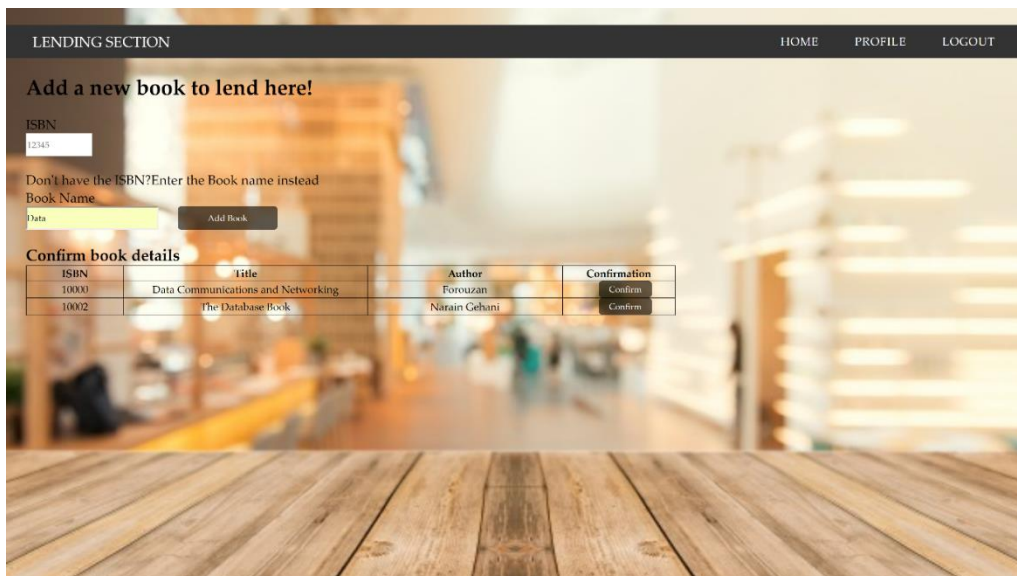


Figure 6.5

Figure 6.5 depicts the add a new book page where the user can search for the information of the book in his possession after which he may upload it.

Reading Section

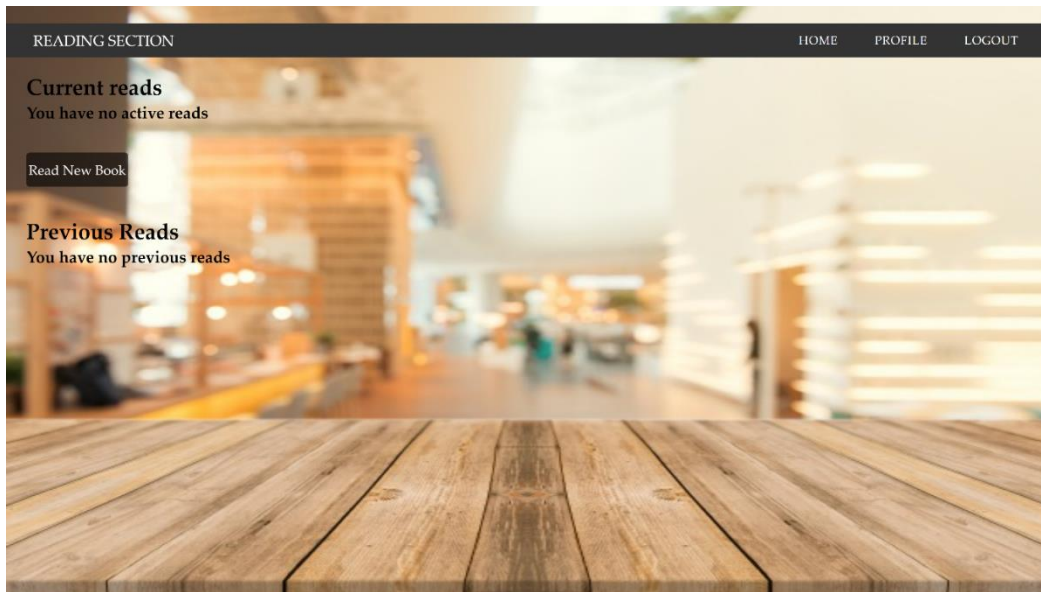


Figure 6.6

Figure 6.6 shows the reading section where the user can view his current reads as well as the books he has previously read.

Request A Book

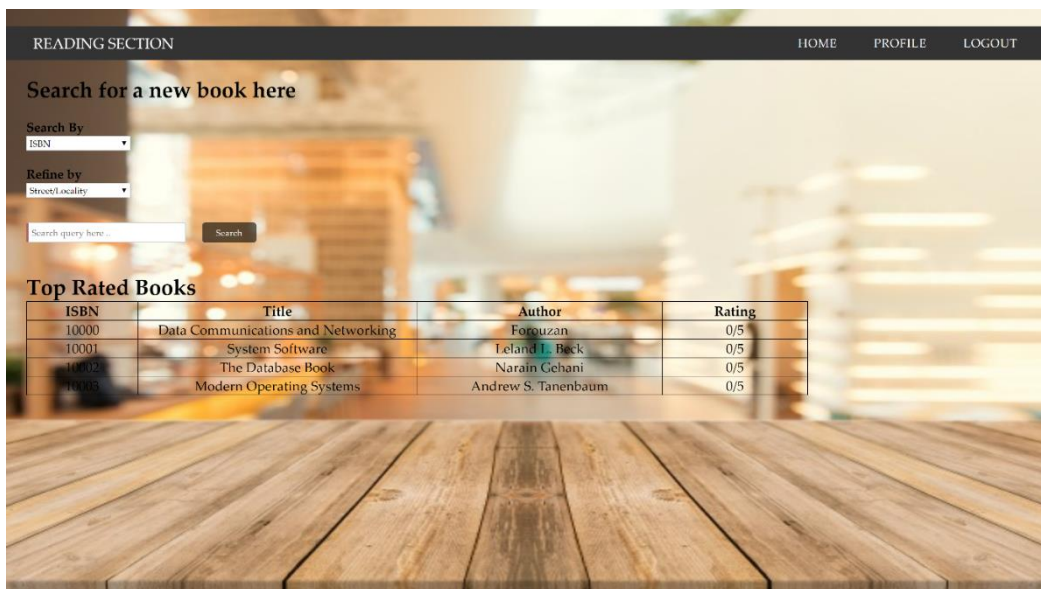


Figure 6.7

Figure 6.7 depicts the read a new book section where the user can search for a new book based on either the book's ISBN or name and may further refine his search based on location. Then request the book from the given list of lenders.

User Profile

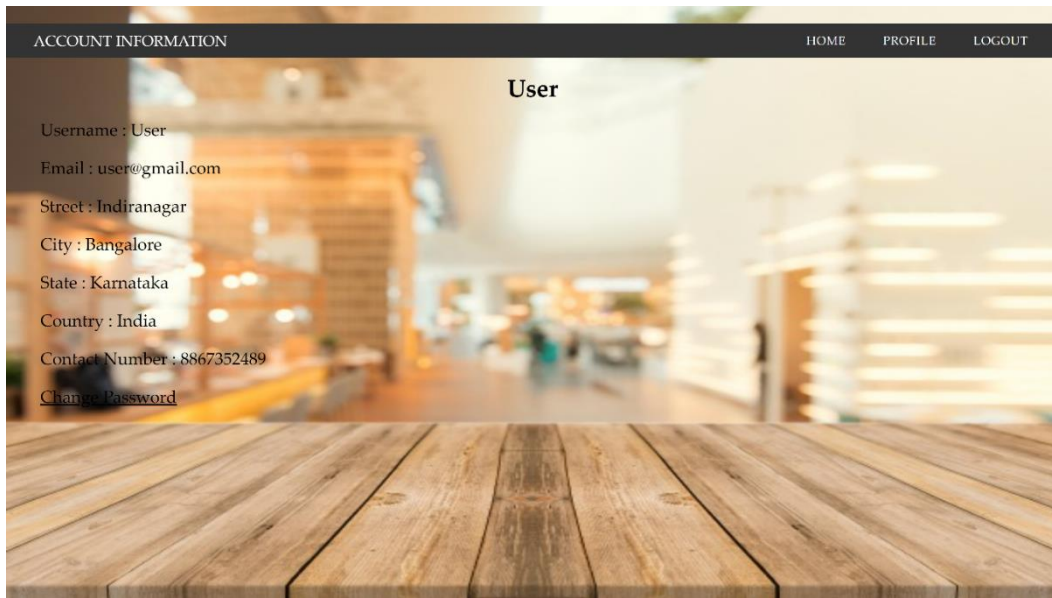


Figure 6.8

Figure 6.8 shows the user's profile where he may view his uploaded details and also change his password to a new one if necessary.

Conclusion and Future Enhancements

The purpose of the website was to create a platform for avid readers to interact with each other and share their books and experiences. With the website, a user can register and then use its services for lending a book in their possession as well as requesting and reading a book that they desire to do the same.

We have strived to achieve most of the goals that we had planned at the beginning of this project.

Future enhancements may include further functionality such as mobile phone and email notifications for the lenders as well as receivers. Also, a better search engine which is dynamic in nature and further information filters can be added.

References

- [1] Narain Gehani, *The Database book: Principles and Practice using MySQL*, Universities press (India) private limited, 2008.
 - [2] Elmasri and Navathe, *Fundamentals of Database System*, Addison-Wesley 5th edition, 2007.
 - [3] Miguel Grinberg, '*Flask Mega Tutorial*', 2017,
<https://blog.miguelgrinberg.com/post/the-flask-mega-tutorial> .
-